Solving distributed and dynamic constraints using an emotional metaphor: Application to the timetabling problem.

Pierre de Loor, Pierre Chevaillier

Solving Distributed and Dynamic Constraints Using an Emotional Metaphor : Application to the Timetabling Problem

P. De Loor* P. Chevaillier†

March 17, 2003

Key words: Distributed and Dynamic Problem, Software agent, Metaphorical Assumption, Emotion, Human Behavior Model.

Abstract

This paper presents a method and it's implementation for solving distributed and dynamic constraints satisfaction problem. In order to improve adaptability and performance, our algorithm is based on agents with autonomous behaviors guided by metaphoric assumptions. Our approach can be distinguished by the following points : The metaphor turns on sociological and emotional criterias without negotiation and memorisation. It tries to copy collective and affective human's behavior during a complex decision making. The agent's model include the notions of affective power, intruder and public mood perception. We have applied this method successfully to the timetabling problem. This paper show formalisation, implementation and first results of this work.

1 Introduction

Our objective is to develop adaptive algorithms for solving distributed and dynamic problem. Recently, some techniques has been developped to solve such king of problem. They are based on distributed operationnal research technics [13] or with multi-agent paradigm [2], [8]. This two classes of approaches start from different models, mechanism and point of view.

- The firts kind of approaches allow a systematic research. Variables control some constraints and communicate their values with the others. Some Values can be refused depending on the constraints. Some algorithms record the refusals. Consequently, in return for time and memories, this approaches are generales and completes []. However, they present some drawback : for complex problems, as allocation or scheduling, the modelisation of the constraints are a hard stage which imply the definition of numbered variables and constraints []. To optimize performances of such algorithms, somes heuristics are used (which do not keep completness properties). They are based on quantitative criterias whithout particular semantic. However, including qualitative criterias, relative to the problem, seem important to improve the performance of the search.

- The second kind of approaches are based of complex autonomous agent including decision and negociation mechanisms. Usually, they are guided by metaphoric assumption like in [11] (auction) or [1] (social insects). The main advantage is relative to the modelisation of the problem by the way of flexible data structures and implicit constraints rather than by a simple set of data and mathematical constraints. However, the backtracking is generally not explicit or forbiden.

2 principe

Our approach is clearly agent oriented. It can be distinguished by the following points : the metaphoric assumption guiding the behavior of the agents turns on sociological and emotionnal criterias [5] : It tries to copy collective and affective human's behavior during a complex decision making. Each agent (representing a human) owns somes goals, skills and a variable which mean it's cognitive power. This variable evolves depending on the perception of differents messages : *requests, denial, proposal, cancellation* (see figure 1). The more important point is that the cognitive power of an agent is altered by messages concerning the agent itself but also by other messages (but with a different strength). Then, each agent perceive the "public mood" of the global system. It is endowed with a self-perception of his part. For example, it can perceive him as an intruder. When it's affective power pass over an emotive threshold, the agent throw a fit : it cancel his commitment (towards other agents) and his affective power is reset. The evolution's rule of the

*Laboratory of Software Engineering, École Nationale d'Ingénieurs de Brest F-29608 Brest cedex – France, deloor@enib.fr

†Laboratory of Software Engineering, École Nationale d'Ingénieurs de Brest F-29608 Brest cedex – France, chevaillier@enib.fr

affective power avoid cycle during the problem solving because it's not always the same agent wich throw a fit but rather the next more intruder agent. This algorithm can be compared to the dynamical hierarchie introduced by [4] but, in our case, this hierarchie is defining for agents by a high level metaphor rather than purely mathematical considerations. Finally, our approaches is dedicated to dynamical's environments. In such environment, variables and constraints can be added or retracted at any time during the search of a solution. The algorithm adapts himself to the current situation. In order to do that, we use two considerations :

- agents don't know the global problem and the set of constraints representing it.
- agent's are unaware of the set (and the number) of the others agents involving in the problem.

In this way, adding agents or constraints implies neither a global restarting of the algorithm nor a modification of it's current mechanisms or structures. Nous prfrons parler de recherche de solution par simulation "anytime", plutot que de rsolution de contraintes.

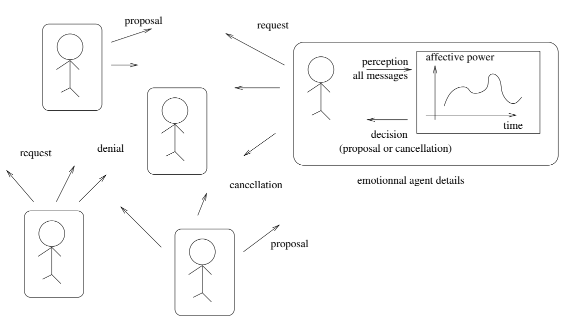

Figure 1: *principe.*

3 Algorithm

3.1 Models

3.1.1 Emotionnal agent's model

An emotional agent A is formalized by an 8-uplet :
$< \psi^A, \alpha_s^A, \alpha_c^A, \rho^A, \gamma^A, S_s^A, R_g^A, P_g^A >$

- $\psi^A \in [0,1]$ is a real meaning the *cognitive power.*
- $\alpha_s^A \in]0,1[$ is a real meaning the *self-sensitiveness rate.*
- $\alpha_c^A \in]0,1[$ is a real meaning the *collective-sensitiveness rate.*
- $\rho^A \in]0,1[$ is a real meaning the *requirement threshold* .
- $\gamma^A \in]0,1[$ is a real meaning the *crisis threshold* ($\gamma^A > \rho^A$).
- S_s^A is the *set of skills.* (symbolic declarations).
- ${R_g}^A$ is the set of *requested goals.*
- ${P_g}^A$ is the set of *personnal goals.*

3.1.2 Goal's model

A Goal g is a 3-uplet $< P^g, S^g, A^g, V^g >$:

- P^g is a first order predicate which express the goal. This symbolic expression is relative to the problem.
- S^g is a skill requiered to achieve or improve the goal (solving P^g).
- A^g is a solution to achive or improve the goal. A solution is a data structure depending of the problem (and can be partial).
- V^g is a boolean meaning the value of P^g.

3.1.3 Message's model

A message M is a 3-uplet $< T^M, E^M, G^M >$:

- $T^M \in \{'request','proposal','cancellation','refusal'\}$ is the message's type.
- E^M is an Agent : the emitter of the message.
- G^M is a goal : the topic of the message .

3.2 Agent's behavior

Figure 2 summarizes the agent's cyclic behavior. Each message processing depends of it's nature. The cognitive power is affected according to the message or the result of the processing. It acts upon the reaction of the agent as a broadcasting of some messages.

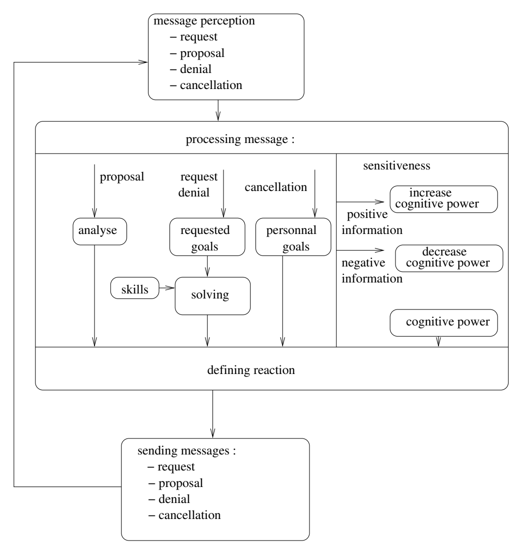

Figure 2: *emotionnal agent newel loop.*

3.2.1 default behavior

The default behavior of an agent A is executed as long as no message is received :

$\forall$ goal $g \in P_g^A$
  if ($V^g == false$)
    $\psi^A = \psi^A + \alpha_s^A * (1 - \psi^A)$

if ($\psi^A > \rho^A$) (requirement)
  $\forall$ goal $g \in P_g^A$
    if ($V^g == false$)
      broadcasting the message $<' request', A, g >$
      $\psi^A = \psi^A - (\alpha_s^A * \psi^A)$

if ($\psi^A > \gamma^A$) (crisis)
  $\forall$ goal $g \in R_g^A$
    broadcasting the message $<' cancellation', A, g >$
  $R_g^A = \phi$
  $\psi^A = 0$

$\psi^A = \psi^A - f(\alpha_s^A, \psi^A)$ (default relaxation)

Summary : each unreached personnal goal incrises the *cognitive power* in proportion to the agent's *self-sensitiveness rate* (α_s^A). If the *cognitive power* exceeds the *requirement threshold*, requests concerning personnal goals are sending. In this case, the *cognitive power* decreases. Such mechanism avoids complex aknowledgments with some answering messages. Indeed, it's the *cognitive power* which regulate the flow of requests. In fact, some policies can be use. For example,the request can depend on a probability depending on the *cognitive power* and the *requiremen threshold*. Only one request for one goal should be broadcasted at one time than requests for all goals. The influence of the policie on the performance of the algorithm must be studied. When the *cognitive power* exceeds the *crisis threshold*, the agent cancels all its commitments towards others agents. In order to do that, it broadcast cancellation's messages relative to all the goal included in it's set of requested goals. In this case, the *cognitive power* is reseted. Therefore, if other agents have a high *cognitive power*, they should enter in the crisis before the previous one. This mechanism can be seen as a local backtrack which goes among the more awkward agents. This backtrack is based on psychological issues and introduce a dynamic hierachie of intruder. To finish, f is a positive function meaning the natural trend to decrising the *cognitive power* when no problem occurs. (The definition of this function will be discribing later). It is basically used to evaluate the convergence's algorithm : when each agent's *cognitive power* decreases, a solution is founded.

3.2.2 messages processing

- processing a message $<' request', A', g >$

if $(A \neq A') \wedge (S^g \notin S_s^A)$
  $\psi^A = \psi^A + \alpha_c^A * (1 - \psi^A)$ (public mood)
else
    solving(P^g)
    if P^g is soluble with a solution s
      $A^g = s$
      $V^g = true$
      $R_g^A = R_g^A \cup g$
      broadcasting the message $<' proposal', A, g >$
      $\psi^A = \psi^A - (\alpha_c^A * \psi^A)$

summary : if the agent has the skill, it attempts to solve the goal whith a *solving* method depending of the problem. The success of this method return a solution in abstract solution's goal form (see 3.1.2 and the exemple).

- processing a message $<' proposal', A', g >$

if $(A \neq A') \wedge \{\exists g' \in P_g^A | P^g == P^{g'}\}$
  $analyse(A^{g'})$
  if ($A^{g'}$ is acceptable)
    $A^g = A^{g'}$
    $V^{g'} = true$
    $\psi^A = \psi^A - (\alpha_s^A * \psi^A)$
  else
      broadcasting the message $<' refusal', A, g >$
else
    $\psi^A = \psi^A - (\alpha_c^A * \psi^A)$ (public mood)

Summary : an agent A is concerned by a proposal if this one is the result of a request from A. Nevertheless, the affective power af an agent is decreased even if it is not concerned by the proposal. It is a part of the perception of the "public mood". Before accepting the proposal, the agent must analyse it with a problem dependent method.

- processing a message $<' refusal', A, g >$

if $(A \neq A') \wedge \{\exists g' \in R_g^A | P^g == P^{g'}\}$
  $\psi^A = \psi^A + (\alpha_s^A * \psi^A)$
  $R_g^A = R_g^A - g'$
else
  $\psi^A = \psi^A + (\alpha_c^A * \psi^A)$ (public mood)

- processing a message $<' cancellation', A, g >$

if $\{\exists g' \in P_g^A | P^g == P^{g'}\}$
  $\psi^A = \psi^A - (\alpha_s^A * \psi^A)$
  $V_g' = false$
else
  $\psi^A = \psi^A - (\alpha_c^A * \psi^A)$ (public mood)

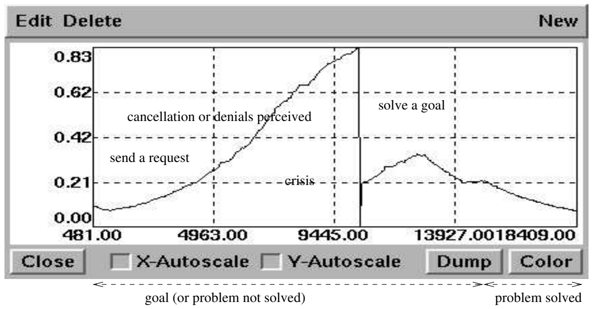

Figure 3: *typical cognitive power evolution.*

3.2.3 Informal justification

The cognitive power depends on the agents' perception and action. In practical term, perception is the reception of broadcasted messages (figure 1). Actions are the research of a solution for a requested goal.

- positives informations : it decreases the cognitive power, and can be devided as following :
  - le : l'agent a russi rsoudre un (ou des) but(s), personnel ou requis.
  - la *dcharge de responsabilit* : l'agent fait une requte car il ne peut rsoudre un but personnel. Dans ce cas, il se repose sur la communaut. Prcisons que contrairement certains protocoles de type *contract net protocol*, l'agent n'opre cette dcharge que pour ses buts personnels et non sur des buts requis.
  - le *succs coopratif* : un agent lui a fait une proposition qui permet de rsoudre un de ses buts personnels.
  - la *crise* : l'agent annule toutes les solutions qu'il avait auparavant proposes. La crise est un vnement particulier qui va provoquer la remise zro de la charge cognitive de l'agent et provoquer une remise en question de la solution en cours de recherche.
- les informations ngatives : elles augmentent la charge cognitive d'un agent. Elles peuvent tre dcomposes de la faon suivante :
  - les *requtes* : toute requte est un signe que "quelque chose n'est pas rsolu au sein du systme multi-agents" et va augmenter la charge cognitive. Si l'agent peut rpondre cette requte, le *succs personnel* conscutif (voir les informations positives) diminue la charge cognitive. S'il n'est pas comptent pour rpondre cette requte, il ne fait rien. S'il est comptent mais qu'il ne peut rsoudre le problme li la requte, il en rsulte un *chec personnel*.
  - les *checs personnels* : l'agent est incapable de rsoudre un problme, qu'il lui soit propre ou pos par un autre agent.
  - les *refus* : l'agent peroit les refus diffuss par les autres agents. Il distingue les refus le concernant directement (annulant une proposition qu'il a faite) et qui accroissent davantage sa charge cognitive que les refus ne le concernant pas. Cependant le fait qu'elle croisse reflte la perception de la "mauvaise ambiance" gnrale au sein du systme multi-agents. Cette proposition part de la constatation que la communication indirecte peut tre un mcanisme fondamental lors de la rsolution de problmes [6].

4 Implementation

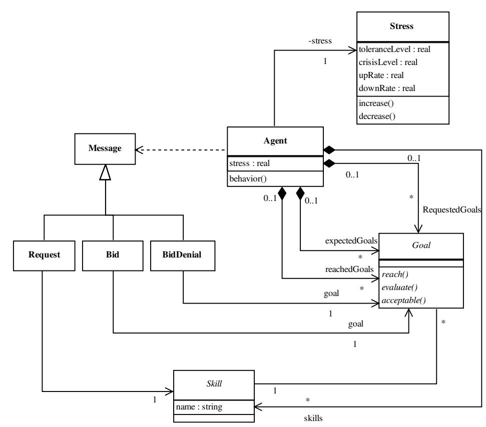

Figure 4: *emotional Agent's class.*

5 Application to the timetabling problem

We applie algorithme au problme de la gnration des emplois du temps rput difficile [12]. Notre solution se diffrencie de celles proposes classiquement [9], [7] par le fait que la recherche est distribue, compltement asynchrone et adaptative. Par contre,contrairement [7] nous ne proposons pas pour l'instant de dmarche d'optimisation qui fait l'objet de nos recherches actuelles. L'article montrera la formalisation de l'algorithme et les rsultats obtenus sur cet exemple implement l'aide du langage oRis [3]. Elaboration d'un langage plus formel de ces changent en broadcast un peu comme propos par [10]

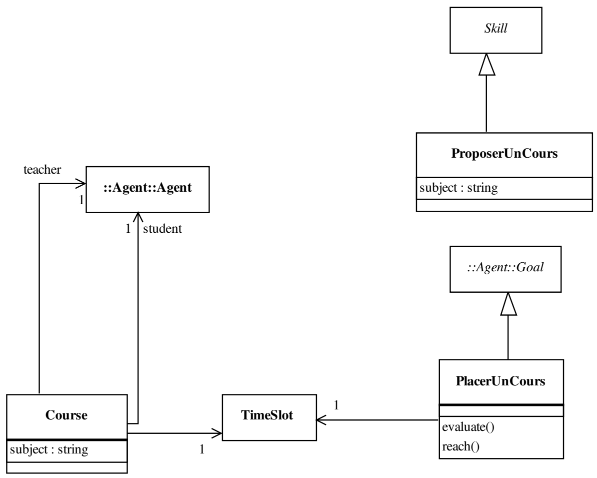

Figure 5: *TimeTable Class.*

5.1 Results

This section will describe the result we obtain. It can include grphical reprsenting the convergence time relative to the problem complexity.

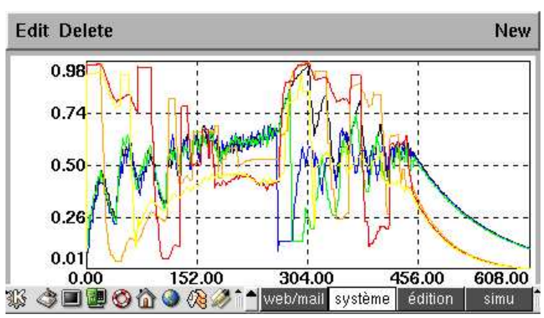

Figure 6: *cognitive power evolution of the agent during the timetabling problem solving.*

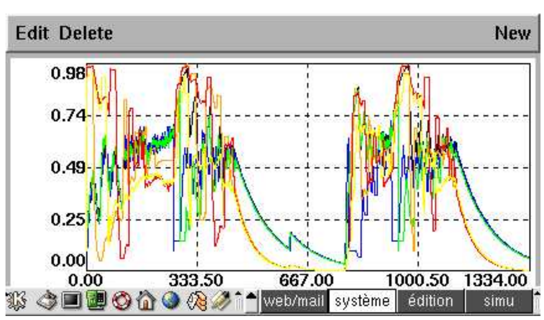

Figure 7: *dynamic adaptation.*

6 conclusion and futur work

References

[1] Eric Bonabeau, Andrej Sobkowski, Guy Theraulaz, and Jean-Luis Deneubourg. Adaptive task allocation inspired by a model of division of labor in social insects. In Dan Lundh, Bjorn Olsson, and Ajit Narayanan, editors, *Biocomputing and Emergent Computation*, pages 36–45. World Scientific, 1997.

[2] E. H. Durfee and J. Rosenschein. Distributed problem solving and multiagent systems: Comparisons and examples. In M. Klein, editor, *Proceedings of the 13th International Workshop on DAI*, pages 94–104, Lake Quinalt, WA, USA, 1994.

[3] Fabrice Harrouet, Jacques Tisseau, Patrick Reignier, and Pierre Chevaillier. oRis : un environnement de simulation interactive multi-agents. *Revue des sciences et technologie de l'information, série Technique et science informatiques (RSTI–TSI)*, 21(4):499–524, 2002.

[4] K. Hirayama and M. Yokoo. An approach to overconstrained distributed constraint satisfaction problems: Distributed hierarchical constraint satisfaction, 2000.

[5] Sternberg R. J. Lautrey J. and Lubart T.I. *Models of Intelligence : International Perspectives.* American Psychological Association, 2002.

[6] B. Pavard J. Dugale and J.L. Soubie. A pragmatic development of a computer simulation of an emergency call centre. In Rose Dieng et al., editor, *Designing Cooperative Systems, Frontiers in Artificial Intelligence.* IOS Press, 2000.

[7] Rainer Staudte Kay Anke and Werner Dilger. Producing and improving time tables by means of constraint and multi-agent systems. In *The AAAI-97 Workshop on Constraints and Agents*, 1997.

[8] Wooldridge M and Jennings N.R. The cooperative problem-solving process. *Journal of Logic and Computation*, 9(4):563–592, 1999.

[9] Hugo Terashima Marn. Combinations of gas and csp strategies for solving the examination timetabling problem, 1998.

[10] T. Torroni and Francesca Toni. Extending a logic based onto-toone negotiation framework to one-to-many negociation. In P. Omicini, A. Petta and R. Tolksdorf, editors, *LNAI*, 2203, pages 105–118, Berlin, 2001. ESAW 2001, Springer-Verlag.

[11] William E. Walsh, Makoto Yokoo, Katsutoshi Hirayama, and Michael P. Wellman. On market-inspired

approaches to propositional satisfiability. In *IJCAI*, pages 1152–1160, 2001.

[12] D. Werra. An introduction to timetabling. *European Journal of Operation Research*, 19:151–162, 1985.

[13] Makoto Yokoo. *Distributed Constraint Satisfaction*. Springer, 2001.